

BAB III

ANALISA DAN PERANCANGAN SISTEM

3.1. Deskripsi Umum

Sistem penentuan kombinasi menu makanan ini dibuat dengan mengimplementasikan algoritma *PSO*, pengguna akan diminta menginputkan data diri seperti nama, tinggi badan, berat badan, jenis kelamin dan sebagainya untuk mendapatkan nilai kebutuhan gizi, yang nantinya akan dijadikan parameter dalam menentukan kombinasi makanan yang tepat. *Output* yang diharapkan dalam sistem ini berupa 3x makan utama dan 3x makan selingan selama sehari.

Penentuan kombinasi menu makanan ini tidak hanya digunakan untuk yang menderita diabetes saja tetapi bisa untuk masyarakat umum juga, anjuran diet diabetes juga bisa digunakan untuk masyarakat umum demi menghindari terkena diabetes.

3.2. Analisa Permasalahan

Diabetes menjadi penyakit yang paling banyak diderita alasannya karena konsumsi makanan yang sembarangan dan tidak teratur, selain itu penyebab diabetes diantaranya keturunan, etnis, usia, obesitas, kurang gerak badan, kehamilan, infeksi, stress dan obat-obatan. Penderita diabetes berisiko sangat tinggi menderita penyakit komplikasi lainnya seperti penyakit jantung, darah tinggi, stroke, gangguan ginjal dan lain-lain.

Resiko ini bisa dicegah dengan control gula darah mengikuti aturan diet yang baik dan sehat. Gula darah akan sangat tergantung dengan makanan yang dimakan dan gizi yang dimiliki makanan tersebut, beberapa variabel berpengaruh dalam control gula darah diantaranya

1. HbA1c yaitu *glycolated hemoglobin* yang berfungsi mengikat gula darah dalam *hemoglobin* selama 2-3 bulan yang dijadikan acuan untuk menentukan apakah pasien tersebut menjalani diet dengan disiplin atau tidak.
2. *C-Reactive Protein* merupakan penanda radang pada infeksi trauma dan sebagainya yang berisiko menimbulkan penyakit jantung, jika ditemukan *hsCRP* tinggi, pasien harus menurunkan berat badan, gula darah.

3. *Trigliserida* merupakan lemak yang dibuat hati dan didapat dari makanan, jadi makan berkalori besar akan menaikkan *trigliserida*
4. *Kolesterol LDL* adalah kolesterol jahat yang bisa menyebabkan penyempitan pembuluh darah, *Kolesterol HDL* adalah kolesterol baik sedangkan *Kolesterol VLDL* merupakan trigliserida yang dikeluarkan dari hati dan menjadi LDL.

Dengan resiko dan penjelasan tersebut diet sehat tepat yang diimbangi dengan pengecekan gula darah sangat efektif untuk mengurangi resiko dan menyeimbangkan gula darah karena diet dapat mengontrol kadar variabel berpengaruh yang terdapat dalam gula darah sehingga resiko tersebut bisa diminimalisir. Berdasarkan hal tersebut dikemukakan sebuah solusi mengontrol gula darah bagi penderita diabetes yaitu dengan pengaturan diet menu makanan berdasarkan kebutuhan gizi dengan parameternya yaitu kalori, protein, karbohidrat dan lemak.

Masalahnya yaitu menentukan menu makanan yang sehat dan bergizi yang sesuai dengan kriteria sangat sulit dilakukan, dikarenakan menentukan kombinasi menu makanan secara manual yang jumlahnya ratusan dengan kandungan gizi yang berbeda-beda akan memakan waktu yang lama dan tidak efektif. Menentukan menu makanan juga tidak boleh sembarangan karena harus sesuai dengan aturan diet untuk penderita diabetes, maka dibutuhkan suatu cara atau metode untuk menentukan kombinasi mana yang paling baik sesuai dengan kebutuhan penderita.

3.3. Usulan Pemecahan Masalah

Penjelasan diatas telah menjelaskan permasalahan yang dikemukakan oleh penulis berkaitan dengan penelitian yang dilakukan, berdasarkan hal tersebut dikemukakan sebuah solusi menentukan kombinasi menu makanan bagi penderita diabetes dengan menggunakan metode optimasi, *Particle Swarm Optimization*. Metode optimasi digunakan untuk memaksimalkan pemilihan kombinasi menu makanan sehingga hasilnya mendekati dengan kebutuhan gizi penderita. Penggunaan algoritma *PSO* diharapkan dapat mengatasi masalah yang dikemukakan dan mendapatkan hasil yang maksimal yaitu memenuhi kebutuhan gizi penderita diabetes atau mempunyai selisih kebutuhan gizi dan rekomendasi menu makanan paling tidak berada diambang batas toleransi $\pm 10\%$ [19].

Dalam menentukan menu makanan yang sesuai dengan penderita diabetes, penulis memilih menu makanan yang sesuai dengan pengaturan diet Departemen Kesehatan Indonesia yang ditunjukkan pada Tabel 2.1, sehingga dipastikan hasil rekomendasi menu makanan merupakan makanan yang sehat dan sesuai aturan diet.

3.4. Analisa Sistem

3.4.1. Deskripsi Sistem

Deskripsi sistem pada aplikasi rekomendasi kombinasi menu makanan ini meliputi 2 aspek, yaitu rekomendasi menu makanan dan pengolahan data. Pada menu status gizi, user dapat langsung melakukan pengecekan rekomendasi menu makanan sesudah memasukkan data diri. User hanya dapat melakukan pengecekan menu makanan dan status gizi sesuai dengan data yang sudah ada.

3.4.2. Analisa Kebutuhan Fungsional

Analisa kebutuhan fungsional untuk memberikan gambaran tentang permasalahan yang terjadi. Permasalahan tersebut adalah sulitnya menentukan menentukan menu makanan harian dengan jumlah gizi yang sesuai dengan kebutuhan penderita diabetes, banyaknya jumlah menu makanan.

Dengan data yang ada maka bisa dijadikan sebuah informasi yang diolah sebagai dasar untuk menentukan menu makanan harian yang sesuai. Data diolah dengan menggunakan algoritma *Particle Swarm Optimization (PSO)*.

3.4.3. Persiapan Data

Data yang digunakan untuk penelitian ini diperoleh dari tabel komposisi bahan makanan Departemen Kesehatan Indonesia serta perangkat lunak NutriClin Departemen Kesehatan dan jurnal terkait bahan makanan. Data tersebut dijadikan acuan sebagai informasi yang mana akan digunakan pada proses optimasi menggunakan algoritma *Particle Swarm Optimization (PSO)*.

3.4.3.1. Sample Data

Berikut adalah contoh isi dari *sample* data untuk menu makanan pokok dan makanan selingan yang disajikan secara berturut-turut pada table 3.1 dan 3.2. Data makanan secara keseluruhan akan disertakan pada **Lampiran 1**.

Tabel 1.1 Sample Data Makanan Pokok

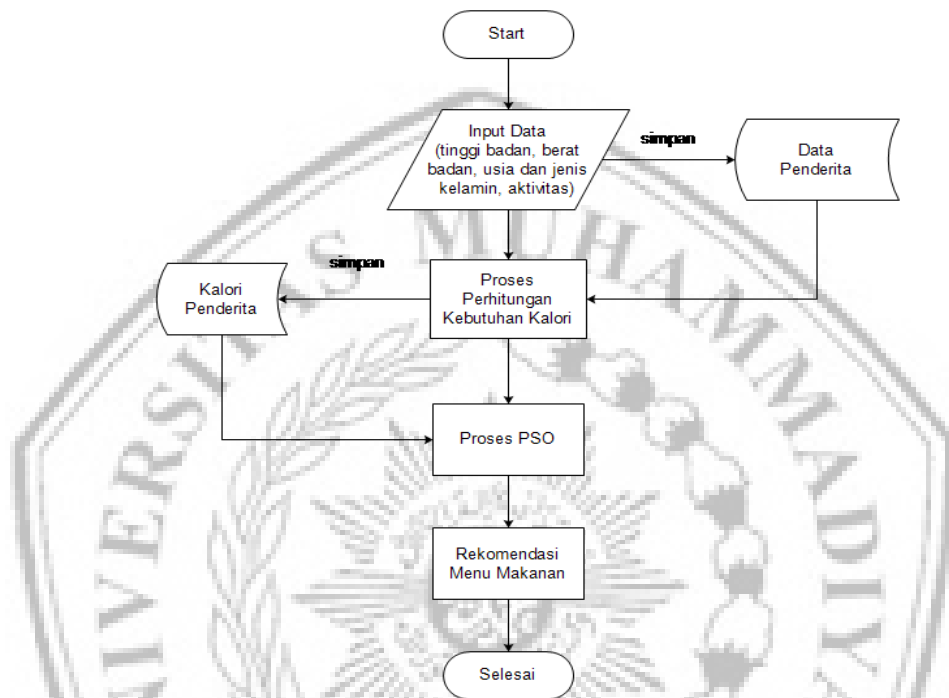
Nama Makanan	Kalori (kal)	Protein (gram)	Lemak (gram)	Karbohidrat (gram)
Bubur Ayam	214	14	7	24
Bubur Kacang Hijau	305	8	3	63
Bubur Ketan Hitam	408	15	1	87
Capcay Seafood	191.2	20.9	8.2	8.8
Coto Makassar	45	2.5	1.2	53
Gado-Gado	137	6.1	3.2	21
Gudeg	160	3.3	9.2	16
Karedok	92	2.2	3.3	14.1
Ketupat Polos	87	2	0	20
Lontong Sayur	391	13	16	36
Nasi Goreng	350.1	6.3	10.5	8
Nasi Goreng + Telur	198	6.4	0	0
Nasi tim	90	1.7	0.02	1.97
Mie Kering Pangsit	409	19	12	55
Mie Kuah Pangsit	217	14	4	31
Mie Ayam	360	5.9	16.2	49.9
Nasi + Ayam semur	321	17.3	9.5	0.3
Nasi + Ayam opor	336.2	16.9	11.1	0.8
Pecel	243	11.1	12.5	31.7

Tabel 3.2 Sample Data Makanan Selingan

Nama Makanan	Kalori (kal)	Protein (gram)	Lemak (gram)	Karbohidrat (gram)
Alpukat	85	0.9	6.5	7.7
Anggur	30	0.5	0.5	6.8
Apel	58	0.3	0.4	14.9
Belimbing	36	0.4	0.4	8.8
Bengkuang	55	1.4	0.2	12.8
Blewah	77	0.45	0.15	18.3
Jambu air	46	0.6	0.2	11.8
Jambu biji	49	0.9	0.3	12.2
Jambu monyet	64	0.7	0.6	15.8
Langsat	56	0.9	0.2	14.3
Lemon	34	0.5	0.8	6.2
Mangga	52	0.7	0	12.3

3.4.4 Flowchart Sistem

Flowchart adalah bagian penting atau langkah awal ketika ingin membangun sebuah sistem. Karena *flowchart* merupakan serangkaian proses yang menggambarkan bagaimana sistem itu bekerja dan berjalan. Berikut adalah *flowchart* secara umum yang disajikan pada gambar 3.1



Gambar 3.1. Diagram alur sistem

Alur program ini dimulai dari interaksi pengguna yang ingin melihat rekomendasi menu makanan yang sesuai dengan kebutuhan tubuhnya. Berikut penjelasan alur dari diagram diatas :

1. Pertama, pengguna memasukkan data diri seperti umur, jenis kelamin, tinggi badan, berat badan dan aktivitas.
2. Selanjutnya, data tadi akan dihitung menggunakan persamaan **Harris Benedict** untuk menentukan kebutuhan gizi harian penderita.
3. Dari hasil perhitungan tersebut didapatkan kebutuhan gizi, yang selanjutnya akan menjadi parameter dalam algoritma *PSO*
4. Jika semua parameter terpenuhi, maka akan masuk ke perhitungan algoritma *PSO*.

5. Setelah dihitung menggunakan algoritma *PSO*, maka sistem akan menampilkan rekomendasi menu makanan sesuai dengan kebutuhan gizi.

3.4.4. Perhitungan Particle Swarm Optimization

1. Hadi terkena penyakit diabetes melitus dan ingin melihat rekomendasi menu makanan yang sesuai dengan kebutuhan hariannya. Hadi mempunyai berat badan 70kg, tinggi badan 175cm, berusia 22 tahun dan beraktivitas futsal 2 kali seminggu.

2. Menentukan Gizi Harian Hadi

- a. Menentukan BMR

$$\begin{aligned}\text{BMR} &= 10 \times \text{Berat Badan (kg)} + 6.25 \times \text{Tinggi Badan (cm)} - 5 \times \\ &\quad \text{Usia} + 5 \\ &= (10 \times 70) + (6.25 \times 175) - (5 \times 22) + 5 \\ &= 700 + 1093.75 - 110 + 5 \\ &= 1688.75\end{aligned}$$

- b. Menentukan nilai faktor aktivitas

Karena Hadi sering berolahraga futsal 2 kali dalam seminggu, maka termasuk yg sedikit aktif. Jadi nilai faktor aktivitasnya adalah sebesar 1,375.

- c. Kebutuhan Kalori / TEE = $1688.75 \times 1,375$

$$= 2322.031$$

- d. Kebutuhan Karbohidrat = $\text{TEE} \times 45\text{-}65\%$

$$\begin{aligned}&= 2322.031 \times 45\text{-}65\% \\ &= 1044.914 \text{ s.d } 1509.320\end{aligned}$$

- e. Kebutuhan Protein = $\text{TEE} \times 10\text{-}20\%$

$$\begin{aligned}&= 2322.031 \times 10\text{-}20\% \\ &= 232.203 \text{ s.d } 464.406\end{aligned}$$

- f. Kebutuhan Lemak = $\text{TEE} \times 20 - 25\%$

$$\begin{aligned}&= 2322.031 \times 20 - 25\% \\ &= 464.406 \text{ s.d } 580.508\end{aligned}$$

3. Masuk ke perhitungan algoritma *Particle Swarm Optimization*, Inisialisasi parameter awal. Parameter yang digunakan penulis pada algoritma PSO ini bersifat tetap, seperti dibawah ini :

1. Parameter $C1 = 1$ dan $C2 = 1$
2. Parameter $r1 = 1$ dan $r2 = 1$
3. Populasi awal sebanyak 40 populasi, posisi awal ditentukan dari urutan dalam daftar tabel menu makanan dan kecepatan ditentukan secara acak.
4. Menu makanan yang ingin didapatkan yaitu 3x makan utama dan 3x makan selingan.
5. Membangkitkan populasi awal sebanyak 4 populasi dan memasukkan menu makanan secara acak sebanyak 6 bahan makanan, pembangkitan populasi awal dilakukan secara berurutan sesuai dengan urutan database menu makanan. 6 bahan makanan tersebut didapatkan dari 3 makanan utama dan 3 makanan selingan.

Tabel 3.2 Pembangkitan populasi awal

	P1	P2	P3	P4
Sarapan	Bubur ayam	Roti isi kacang	Nasi + Dadar telur ayam	Nasi + Roll Ayam
Selingan	Apel	Biskuit	Jus Alpukat	Anggur
Makan Siang	Nasi + Ayam Panggang	Soto Ayam	Mie Ayam	Nasi + Mujair acar kuning
Selingan	Pudding	Alpukat	Pepaya	Nanas
Makan Malam	Nasi + Gurami Asam manis	Nasi + Telur Ceplok	Nasi + Rendang	Nasi + Rendang
Selingan	Anggur	Salad Buah	Kue kering	Jus Wortel
Total	Kalori = 1097 Protein = 61.01 Lemak = 24.73 KH = 59.7	Kalori = 1464 Protein = 54.01 Lemak = 51.32 KH = 153.5	Kalori = 1308 Protein = 56 Lemak = 47.6 KH = 82.83	Kalori = 1537 Protein = 97.7 Lemak = 43.3 KH = 127.7

Pembangkitan populasi awal dilanjutkan dengan mengkalkulasi total kebutuhan gizi berupa kalori, protein, lemak dan karbohidrat pada tiap partikel yang ada.

6. Melakukan perhitungan *fitness* tiap *partikel*

Fungsi *fitness* digunakan untuk mengukur tingkat kebaikan atau kesesuaian (*fitness*) suatu solusi dengan solusi yang dicari. Perhitungan penalti tiap partikel menggunakan persamaan 8 dengan nilai kebutuhan gizi dan kandungan gizi sebagai parameter.

$$\begin{aligned} 1. \text{ Penalti P1} &= |2322.031 - 1097| + |1044.914 - 59.7| + |232.203 - 61.01| + |464.406 - 24.73| \\ &= 2821,286 \end{aligned}$$

$$\begin{aligned} \text{Fitness P1} &= \frac{1}{\text{Penalti}} \text{Const} + \text{Constraint} \\ &= \frac{1}{2821,286} 10000 + 1 = 4,54 \end{aligned}$$

$$\begin{aligned} 2. \text{ Penalti P2} &= |2322.031 - 1464| + |1044.914 - 153.5| + |232.203 - 54.01| + |464.406 - 51.32| \\ &= 2340,724 \end{aligned}$$

$$\begin{aligned} \text{Fitness P2} &= \frac{1}{\text{Penalti}} \text{Const} + \text{Constraint} \\ &= \frac{1}{2340,724} 10000 + 1 = 5,27 \end{aligned}$$

$$\begin{aligned} 3. \text{ Penalti P3} &= |2322.031 - 1308| + |1044.914 - 82.83| + |232.203 - 56| + |464.406 - 47.6| \\ &= 2569,124 \end{aligned}$$

$$\begin{aligned} \text{Fitness P3} &= \frac{1}{\text{Penalti}} \text{Const} + \text{Constraint} \\ &= \frac{1}{2569,124} 10000 + 1 = 4,89 \end{aligned}$$

$$\begin{aligned} 4. \text{ Penalti P4} &= |2322.031 - 1537| + |1044.914 - 127.7| + |232.203 - 97.7| + |464.406 - 43.3| \\ &= 2257,854 \end{aligned}$$

$$\begin{aligned} \text{Fitness P4} &= \frac{1}{\text{Penalti}} \text{Const} + \text{Constraint} \\ &= \frac{1}{2257,854} 10000 + 1 = 5,43 \end{aligned}$$

Setiap pelanggaran constraint yang dilakukan akan memberikan nilai -1 pada nilai *Constraint* dan akan diakumulasi sesuai dengan jumlah pelanggaran yang dilakukan.

7. Menentukan *pBest*

Partikel dengan nilai *fitness* paling besar dari partikel-partikel lain dalam satu iterasi disimpan sebagai *local best particle* (*pBest*).

Tabel 3.4 Penentuan *pBest* berdasarkan nilai *fitness*

F (partikel, iterasi)	Posisi	Fitness
F (1,0)	[1, 3, 47, 31, 54, 2]	4,54
F (2,0)	[30, 34, 21, 1, 45, 33]	5,27
F (3,0)	[41, 25, 16, 16, 51, 35]	4,89
F (4,0)	[71, 2, 56, 15, 51, 26]	5,43

Karena nilai *fitness* yang paling besar dari keseluruhan partikel adalah 5,43 dan terdapat pada partikel ke-4 maka partikel ke-4 **F(4,0)** ditentukan sebagai *pBest*.

8. Menentukan *gBest*

Nilai *global best* pada iterasi pertama adalah sama dengan nilai *local best* pada iterasi pertama, kemudian untuk iterasi selanjutnya dilakukan *update*. Dan disimpan sebagai *global best particle*. Karena masih *iterasi* awal, maka *pBest* dan *gBest* ada pada *partikel* yang sama yaitu **F(4,0)**.

9. Melakukan Update *Velocity* / Kecepatan, dengan menggunakan persamaan dibawah ini dengan parameter nilai *Velocity* awal = 0.

$$V_{id}^{t+1} = V_{id} + C^1 r^1 (P_{id} - X_{id}) + C^2 r^2 (G_{id} - X_{id})$$

$$\begin{aligned} \text{a. } V(1,1) &= 0 + 1 \cdot 1 ([71, 2, 56, 15, 51, 26] - [1, 3, 47, 31, 54, 2]) + 1 \cdot 1 \\ &\quad ([71, 2, 56, 15, 51, 26] - [1, 3, 47, 31, 54, 2]) \\ &= [70, -1, 9, -16, -3, 24] + [70, -1, 9, -16, -3, 24] \\ &= [140, -2, 18, -32, -6, 48] \end{aligned}$$

$$\begin{aligned} \text{b. } V(2,1) &= 0 + 1 \cdot 1 ([71, 2, 56, 15, 51, 26] - [30, 34, 21, 1, 45, 33]) + \\ &\quad 1 \cdot 1 ([71, 2, 56, 15, 51, 26] - [30, 34, 21, 1, 45, 33]) \\ &= [41, -32, 35, 14, 6, 7] + [41, -32, 35, 14, 6, 7] \\ &= [82, -64, 70, 28, 12, 14] \end{aligned}$$

$$\begin{aligned} \text{c. } V(3,1) &= 0 + 1 \cdot 1 ([71, 2, 56, 15, 51, 26] - [41, 25, 16, 16, 51, 35]) \\ &\quad + 1 \cdot 1 ([71, 2, 56, 15, 51, 26] - [41, 25, 16, 16, 51, 35]) \\ &= [30, -23, 40, -1, 0, -9] + [30, -23, 40, -1, 0, -9] \end{aligned}$$

$$= [60, -46, 80, -2, 0, 18]$$

$$\begin{aligned} \text{d. } V(4,1) &= 0 + 1 \cdot 1 ([71, 2, 56, 15, 51, 26] - [71, 2, 56, 15, 51, 26]) \\ &\quad + ([71, 2, 56, 15, 51, 26] - [71, 2, 56, 15, 51, 26]) \\ &= 0 \end{aligned}$$

Proses update *velocity* baru menggunakan parameter nilai *velocity* yang lama, C_1 , C_2 , *Local Best*, *Global Best*, bilangan random r_1 dan r_2 dalam interval $[0,1]$.

10. Melakukan Update *Position*, dengan menggunakan persamaan dibawah ini.

$$X_{id}^{t+1} = X_{id} + V_{id}^{t+1}$$

$$\begin{aligned} \text{a. } P(1,1) &= F(1,0) + V(1,1) \\ &= |[1, 3, 47, 31, 54, 2] + [140, -2, 18, -32, -6, 48]| \\ &= [141, 1, 65, 1, 48, 50] \\ \text{b. } P(2,1) &= F(2,0) + V(2,1) \\ &= |[30, 34, 21, 1, 45, 33] + [82, -64, 70, 28, 12, 14]| \\ &= [112, 30, 91, 29, 57, 19] \\ \text{c. } P(3,1) &= F(3,0) + V(3,1) \\ &= |[41, 25, 16, 16, 51, 35] + [60, -46, 80, -2, 0, 18]| \\ &= [101, 21, 96, 14, 51, 17] \\ \text{d. } P(4,1) &= F(4,0) + V(4,1) \\ &= [71, 2, 56, 15, 51, 26] + 0 \\ &= [71, 2, 56, 15, 51, 26] \end{aligned}$$

11. Sehingga didapatkan menu makanan baru sesuai dengan urutan pada daftar tabel menu makanan.

- a. $P(1,1) = [141, 1, 65, 1, 48, 50]$
[Nasi Ayam goreng + Sayur tahu sosin, Alpukat, Nasi Ayam bakar + Sayur bening kecambah dan jagung, Alpukat, Nasi Abon Ikan + Ketimun, Kentang Panggang Isi Ayam]
- b. $P(2,1) = [112, 30, 91, 29, 57, 19]$
[Nasi Ayam Bumbu Kuning + Sayur bayam campur oyong, Jagung rebus, Nasi + **Ayam** Bumbu Kuning, Jeruk Manis, Nasi Abon Ikan + Sayur kacang merah, Dodol Bali]

c. $P(3,1) = [101, 21, 96, 14, 51, 17]$

[Nasi Ayam Bumbu Kuning + Daun kacang panjang, Dodol Galamay, Nasi Ayam Bumbu Kuning Sayur bening labu siam wortel, Bolu Kukus, Nasi Abon Ikan + Sayur Asem, Dodol]

d. $P(4,1) = [71, 2, 56, 15, 51, 26]$

[Nasi + Roll Ayam, Anggur, Nasi + Mujair acar kuning, Nanas, Nasi + Rendang, Jus Wortel]

12. Mengulangi langkah 4-10 sampai *iterasi* hingga kondisi berhenti tercapai dengan hasil berupa *partikel* terbaik dari seluruh *iterasi*.

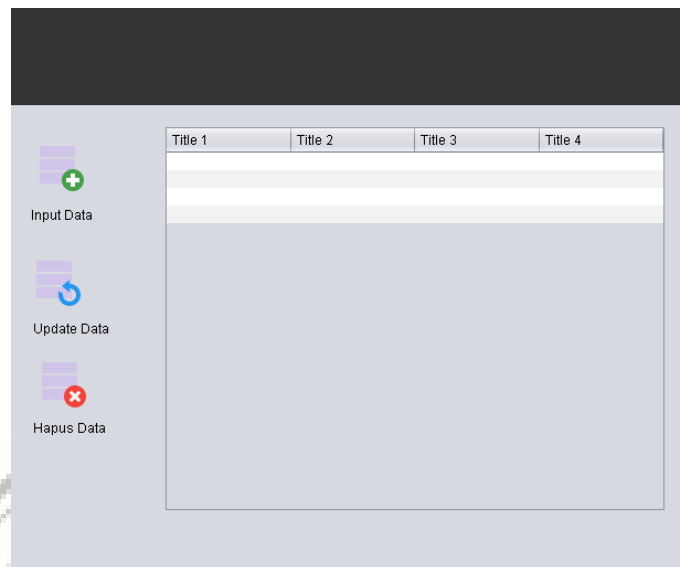
3.5. Rancangan Antarmuka

Rancangan antarmuka atau *interface* merupakan sebuah rancangan dengan gambar yang menunjukkan bagian – bagian seperti tampilan awal, menu dan fitur serta fungsi – fungsi lain yang terkandung didalam sistem yang akan dibangun. Rancangan ini juga berhubungan dengan alur kerja sistem karena menggambarkan secara *visual* bagaimana sistem akan dibuat dan bagaimana *output* yang akan dihasilkan oleh sistem.

Gambar 3.2. Menu Status Gizi

Rancangan antarmuka menu status gizi digunakan untuk user yang bertujuan mengetahui status gizi atau kebutuhan gizi harian dengan terlebih dahulu

memasukkan data diri ke dalam sistem yang nanti akan menampilkan kebutuhan gizi dan rekomendasi menu makanan harian sesuai dengan kebutuhan user.



Gambar 3.3. Menu Data Makanan

Rancangan antarmuka menu data makanan merupakan menu yang digunakan untuk mengelola data menu makanan, seperti menginput data makanan yang baru, mengedit maupun menghapus data yang tidak valid.